הגנה ברשתות – תרגיל בית 2

אורי מינץ 314616897

שאלה 1 – התקפות על רשתות ונוזקות

סעיף 1

a. התקפת ARP spoofing היא סוג של תקיפת רשת על ידי תקיפה של פרוטוקול ה-ARP. התקיפה מתבצעת כך: תחנה כלשהי מבקשת לדעת את כתובת ה-MAC של תחנה אחרת לפי כתובת ה-IP שלה, ההודעה הזאת עוברת ברשת ה-LAN וכל מחשב שאינו בעל כתובת ה-IP הנכונה מתעלם ממנה, אבל התוקף שולח <u>ראשון</u> את התשובה ומביא את כתובת ה-MAC שלו למרות שה-IP אינו מתאים לו. כיוון שהתרגום נקבע לפי התגובה הראשונה, אם ההודעה של התוקף מגיעה קודם הכתובת ה-IP תשויך אליו במקום למחשב האמיתי. בכך התוקף יוכל להתחזות למחשב הקורבן, להאזין למידע שנשלח אליו ואפילו לשנות אותו.

b. התוקף יכול לשלוח באופן תמידי למחשב שהולך לבצע שאילתת ARP הודעות עם ה-IP אליו הוא רוצה להתחזות ועם כתובת ה-MAC שלו, כאשר המחשב יבצע את השאילתת ARP יש סיכוי גדול שהוא מיד יקבל את התשובה מהתוקף לפני שהבקשה תגיע בכלל למחשב עם הכתובת MAC המתאים לכתובת ה-IP.

c. כדי למנוע מתקפות ARP מנהל מערכת יכול לבצע להחזיק מערך קבוע של כל תרגומי ה-IP ל-MAC במערכת ואז אם רוצים לבצע הוספה של מחשב חדש מוספים למערך באופן ידני. כאשר תתבצע בקשה של תרגום IP ל-MAC ב-LAN יוחזר הארך המתאים במערך.

סעיף 2

מומלץ להתקין במהירות security updates כיוון שכאשר מתפרסם עדכון כזה, הוא בעצם מראה לתוקפים חולשה שהייתה קיימת במערכת ההפעלה לפני התיקון. תוקפים אילו יכולים לנסות לבצעה הנדסה לאחור על הקובץ ולנסות לתקוף את כל המחשבים שעדיין לא ביצעו את העדכון של התוכנה. הדוגמה בתרגול היא בלסטר.

סעיף 3

a. ב-2 בנובמבר 1988 היה כינוס של מומחי מערכות Unix ב-MIT, באותו אדם בשם Robert Tappan Morris הפעיל את התולעת האינטרנט שהוא כתב שפגעה במחשבי Unix, תולעת זאת הדביקה כ-10% מ-60,000 המחשבים שהיו ברשת. התולעת מחפשת מחשבים ברשת אחר כך היא חודרת אל המחשבים (היא מנסה בכמה דרכים שונות), לאחר מבן היא מדביקה את המחשב, יוצרת תולעת חדשה ומוחקת את כל הקבצים שנוצרו בדיסק ורצה רק בזיכרון. דבר זה מעמיס על הזיכרון (כיוון שהיא נדבקת בעוד תולעים שאחד מתוך שבעה ועותקים בממוצע לא מתאבד ובכך מספר התולעים על המחשב בזיכרון עולים) של המערכת וכיוון שכל כמה דקות מתבצע fork התהליך נמצא תמיד בתחילת סדר תור העדיפויות. נוצר מצב של שאין מקום יותר בזיכרון (בגלל התולעת והשימוש של הסטודנט במחשב) המחשב קרס.

b. כיוון שהתולעת שמורה רק בזיכרון (הנדיף) בכל פעם שהמחשב נכבה היא נמחקת ולכן המחשב לא יקרוס מייד אלא רק לאחר זמן שבוא הוא ידבק בעוד תולעים והוא יקרוס מחוסר בזיכרון.

c. הסטודנט יכול לנתק את המחשב מהרשת (לאחר שהתולעת נמחקת כתוצאה מקריסת המחשב), לסיים את העבודה כאשר הוא מנותק וללא תולעים. לאחר שהוא מסיים ושמר את הקובץ בדיסק והוא יכול לחבר חזרה את המחשב ולרשת ומהר לשלוח את המסמך תשובות שלו.

סעיף 4

a. חולשת buffer overflow היא חולשה הנגרמת משימוש בפונקציה gets (או scanf וכו') שקולטת מהמשתמש קלט וכותבת אותו על ה-stack. כך המשתמש יוכל לדרוס תוכן שהיה ב-stack ולכתוב בה כל דבר שרוצים. תולעת יכולה לנצל חולשה זו על ידי דריסה של הערך חזרה מהפונקציה שקראה ל-gets ולשים בה כתובת אחרת, כתובת זו תהיה תחילת הקוד של התולעת שגם הוא נכתב בעזרת gets (כלומר כותבים גם את הקוד וגם את מיקום הקוד במקום המתאים).

קוד התולעת יכיל קוד מכונה שמבצע execve('/warm',0,0) ואז התוכנה תתחיל לבצע את קוד התולעת.

b. התולעת שכתב Robert Tappan Morris (תולעת האינטרנט) השתמשה ב-buffer overflow כדי לבצע חדירה למחשבים (שפגעה במחשבי Unix ב-2 בנובמבר 1988).
דוגמה נוספת היא שימוש של תולעת Blaster בחולשה MS03-026 (חולשה מטיפוס buffer overflow).

c. התולעת שכתב Robert Tappan Morris (תולעת האינטרנט) ניצלה את החולשה הזאת ב-finger תוכנית שמאפשרת לקבל מידע על משתמש. אם התוכנית מקבלת קלט מהמשתמש וקוראת אותו בעזרת הפקודה gets.כאשר gets נקראת בתוכנית של finger נכתב אליה מחרוזת של קוד התולעת שכולל ביצוע execve המפעיל את התולעת, בעזרת דריסת ערך החזרה של הפונקציה הקוד הזה מופעל.

סעיף 5

היה צריך שכל העותקים יוכלו להתאבד, כלומר שלא יהיה מצב שאחד מתוך שבעה עותקים (בממוצע) אף פעם לא מתאבד.

סעיף 6

Syn Attack היא סוג של תקיפה DDoS שבה התוקף מציף את השרת בהרבה בקשות קישור של TCP אבל לא מסיים לבצע את הקישור של החיבור בעזרת שליחת ACK. לכן השרת ימשיך לנסות לקבל את ה-ACK על ידי שליחה חוזרת של ACK למחשב שמבצע את התקיפה וכיוון שהתור של הלקוחות מוגבל, הוא מתמלא בבקשות של התוקף ובכך השרת לא יכול לתת שירות ללקוחות אמיתיים שצריכים את השירות שלו כיוון שלא יהיו משאבים (זיכרון, רוכב פס תקשורת וזמן מעבד). בדרך כלל יש הגנה על שליחה מ-IP יחיד לכן בדרך כלל ההודעות ישלחו מ-IP שונים.

שאלה 2 – TCP/IP

סעיף 1

תקשורת באינטרנט מבוצעת ב-5 שכבות שונות, כל שכבה "רוכבת" על גבי השכבה מתחתיה.
השכבה הראשונה היא השכבה הפיזית, זאת השכבה שמתקשרת בביטים פיסיים כמו חיווט ושידור רדיו ואין פרוטוקולים בשכבה הפיזית.
מעלה יש את שכבה השנייה, שכבת ה-MAC שאחראית על תקשורת בין שכנים, דוגמה לפרוטוקול בשכבה זו הוא הרשתות Ethernet.
שכבה שלוש נקראת שכבת הרשת ותפקידה הוא העברת חבילות בין מחשבים מרוחקים. פרוטוקול לדוגמה הוא IP4.
מעליה יש את שכבה ארבע שנקראת שכבת התובלה. תפקידה הוא ווידוא העברה תקינה של המידע ובקצב מתאים, העברת תקשורת בין אפליקציות במחשבים שונים, וסיפוק שירותים לכל האפליקציות במכונה. דוגמה לפרוטוקול הוא TCP.
השכבה הגבוהה ביותר היא שכבה חמש, שכבת האפליקציה שתפקידה ביצוע תקשורת ברמת האפליקציה לפי הצרכים שלה. דוגמה לפרוטוקול הוא https.

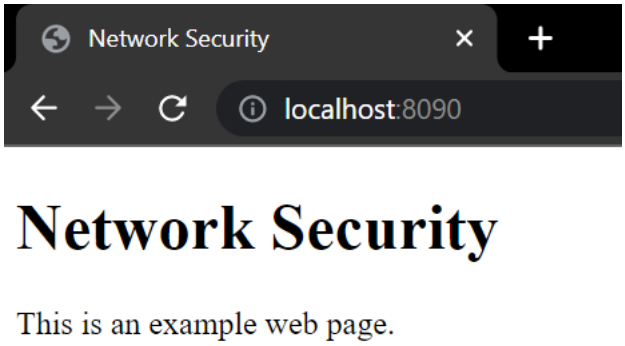
סעיף 2

הקוד מבוסס על socket — Low-level networking interface — Python 3.11.3 documentation

```
PS C:\Users\lenovo\OneDrive\Documents\programing_projercts\Network Security\hw2> python client.py www.google.com 80

HTTP/1.1 200 OK
Date: Mon, 01 May 2023 18:28:42 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=ISO-8859-1
Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-_srGm-WRD0yWufWazoD77g' 's
trict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https: http:;report-uri https://csp.withgoogle.com/csp
/gws/other-hp
P3P: CP="This is not a P3P policy! See g.co/p3phelp for more info."
Server: gws
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Set-Cookie: 1P_JAR=2023-05-01-18; expires=Wed, 31-May-2023 18:28:42 GMT; path=/; domain=.google.com; Secure
Set-Cookie: AEC=AUEFqZdJ_0QQHLHqwZaT6fqfKwjzIAMzy4h5MluEaGyHQiH5IF7YMZokI_s; expires=Sat, 28-Oct-2023 18:28:42 GMT;
 path=/; domain=.google.com; Secure; HttpOnly; SameSite=lax
Set-Cookie: NID=511=iuvnArf2xg3udGm2e8ZgXAr7bIWaaiEx-z6WY7tdmsyka-lhmqgk7QJ8a7eXZTPmab4cbcBVTZqzro4eojp1SnhNhzlnMn8
zp6JPmf1yNq1NDZnooDoKWPb8xUjQrmpMgKFky6wcK9Si5QDatk7v4VlcydCAyDI-0DI5kE3QaKY; expires=Tue, 31-Oct-20
```

סעיף 3



סעיף 4

```
PS C:\Users\lenovo\OneDrive\Documents\programing_projercts\Network Security\hw2> python client.py localhost 8080
HTTP/1.1 200 OK
Content-Type: text/html

<!DOCTYPE html>

<html>
<head>
    <title>Network Security</title>
</head>

<body>
    <h1>Network Security</h1>


    <p>This is an example web page.</p>
</body>
</html>
```

סעיף 5

a. הפונקציה bind מקשרת בין מספר פורט וכתובת IP ל-socket. השתמשתי בה ב-server.py לאחר יצירת ה-socket.
b. הפונקציה listen מכניסה את ה-socket למצב האזנה, היא מקבלת בתור ארגומנט את אורך תור החיבור המקסימלי שהתור יכול לקבל. השתמשתי בה ב-server מיד אחרי bind.
c. הפונקציה connect מבצעת את הקישור בין ה-socket של הלקוח לשרת לפי המספר IP וה-port. השתמשתי בה ב-client.
d. הפונקציה accept מאפשרת קבלה של תקשורת לאחר ביצוע של listen (היא מקבלת את הפורט של ה-client)
השתמשי בה ב-server כדי לקבל תקשורת מהלקוחות.
e. הפונקציות הללו מאפשרות קבלה ושליחה של מידע בהתאמה. השתמשתי ב-recv בלקוח כדי לקרוא את התשובה ששלח השרת והשתמשתי ב-sendall (פונקציה זהה ל-send רק שהיא מבטיחה שכל המידע נשלח) גם בשרת וב-send בלקוח.

שאלה 3: חומות אש

סעיף 1

אם לעובד יש הרשאות מנהל (root), הוא יכול ליזום התחברות לשרת ה-DB דרך מספר פורט נמוך מ-1024, ואז לפי הטבלה, הוא לא מתאים לשורה השלישית (DB out) כיוון שמספר הפורט שלו לא גדול מ-1023 והוא לא מתאים לאף שורה לפני 3 ולא ל-4 כיוון שהוא שולח לפורט 80 והכיוון הוא out. לכן השורה המתאימה תהייה Default והיא מאפשרת חיבור.

סעיף 2

Rule	Direction	Src. Addr	Dst. Addr	Protocol	Src. Port	Dst. Port	ACK	Action
spoof1	in	TLV	external	any	any	any	any	Deny

spoof2	out	external	TLV	any	any	any	any	Deny
HTTP out	out	TLV	any	TCP	>1023	443	any	Allow
HTTP in	int	any	TLV	TCP	443	>1023	yes	Allow
DB out	out	M	DB	any	>1023	80	any	Allow
DB in	in	DB	M	any	80	>1023	any	Allow
default	any	any	any	any	any	any	any	Deny

סעיף 3

- העובד יבצע IP Spoofing למחשב של המנהל, כיוון ששניהם נמצאים ברשת הפנימית חומת האש לא תמנע את זה, הוא ישלח ל-DB פקודה IP Amount 1 ב-UDP עם ה-IP של המנהל וכיוון שהדבר היחידי שהשרת עושה זה מוודא זה את ה-IP העובד יצליח להשיג העלאת שכר.
- בעזרת ARP spoofing העובד יוכל לגרום לנתב לחשוב שהעובד משויך לכתובת IP המתאימה למנהל (ניתן לעשות זאת כיוון שהם על אותו LAN). אחר כך העובד ישלח הודעה ל-DB כאשר הוא כותב בהודעת החיבור של ה-TCP את ה-IP של המנהל שאליו הוא מתחזה. ההודעה תעבור את ה-firewall ותגיע לשרת שימשיך את לחיצת הידיים. כיוון שהנתב חושב שה-IP של העובד הוא של המנהל, הוא יעביר את המשך לחיצת הידיים אליו והוא ימשיך אותה עד ליצירת קשר TCP. לאחר מכן הוא יוכל לשלוח את ההודעה ID Level 2 ולקבל העלאה בדרגה (כאשר הוא ממשיך להתחזות ל-IP של המנהל שלו).